

Data oddania: _____

Ocena: _____

Tomasz Trębski 158065

Sebastian Jakowski 165415

Andrzej Prokopczyk 165502

Zadanie 1: Piętnastka

1. Cel

Celem zadania była implementacja dwóch programów:

- rozwiązującego układankę "piętnastka"
- wizualizującego kolejne kroki rozwiązania uzyskanego w pierwszym programie

2. Strategie przeszukiwania

W naszym programie zgodnie z wymaganiami wykorzystaliśmy następujące metody przeszukiwania:

1. **DFS**: przeszukiwanie w głąb polega na badaniu wszystkich krawędzi wychodzących z podanego wierzchołka. Po zbadaniu wszystkich krawędzi wychodzących z danego wierzchołka algorytm powraca do wierzchołka, z którego dany wierzchołek został odwiedzony. Kroki algorytmu:
 - a) Oznacz aktualny węzeł jako odwiedzony
 - b) Dla każdego sąsiada aktualnego węzła:
 - i. Sprawdź czy był on już odwiedzony
 - ii. Jeśli nie był, rozpocznij procedurę DFS dla tego węzła
2. **IDFS**: Polega na zbadaniu wszystkich węzłów w odległości d od stanu początkowego, w przypadku braku rozwiązania głębokość przeszukiwania jest zwiększana i całe przeszukiwanie rozpoczyna się od początku. Algorytm:
 - a) Przeszukaj wszystkie węzły w odległości d od stanu początkowego

- i. Jeśli stan końcowy został osiągnięty - zakończ
 - b) W przeciwnym wypadku zwiększ głębokość przeszukiwania i rozpocznij procedurę od nowa
3. **BFS**: przeszukiwania grafu wszerek rozpoczyna się od zadanego wierzchołka s , a kończy w momencie osiągnięcia grafu końcowego k . Począwszy od wierzchołka początkowego odwiedzane są wszystkie wierzchołki, które można osiągnąć z danego wierzchołka.

Algorytm:

- a) Dodaj wierzchołek startowy do kolejki.
- b) Pobierz element z kolejki.
 - Jeśli element z kolejki równy szukanemu, koniec algorytmu.
 - W innym wypadku pobierz wszystkich możliwych sąsiadów i wprowadź ich do kolejki.
 - Jeśli któryś z sąsiadów jest elementem końcowym, koniec algorytmu.
- c) Jeśli kolejka pusta, wszystkie węzły w grafie zostały odwiedzone, koniec algorytmu.
- d) W innym wypadku, powrót do kroku 2.

Przygotowany przez nasz algorytm BFS, praktycznie zawsze kończy się w wyniku znalezienia w jednym z następników danego węzła, węzła końcowego. Dlatego też, krok polegający na sprawdzeniu rozmiaru kolejki nie jest nigdy osiągalny.

4. **A-Star**: jest kolejnym z algorytmów szukających ścieżki w grafie, z tą różnicą, że chodzi zawsze o najkrótszą ścieżkę od węzła wejściowego do końcowego. **A*** zaczyna od sprawdzenia węzłów, które są sąsiadami danego węzła i które nie zostały jeszcze odwiedzone. Jego dużą zaletą jest to, że nie posiada on cech algorytmu zachłannego, ponieważ przy wyborze dalszej drogi uwzględnia on koszt dotarcia do następnego węzła, jak i do węzła końcowego. Warto wprowadzić więc 3 parametry krawędzi łączącej dwa węzły:

- **G** - koszt ścieżki prowadzącej z punktu startu do aktualnej, rozpatrywanej pozycji w przestrzeni. Innymi słowy jest to całkowity koszt już wygenerowanej ścieżki;
- **H** - wyliczona na podstawie przyjętej heurystyki, szacunkowy koszt dotarcia do kolejnego węzła lub do węzła końcowego;
- **F** - całkowity koszt danej krawędzi

Ostatecznie, aby algorytm poprawnie realizował swoje zadanie, należy zdefiniować 3 kolekcje:

- **openSet** - zbiór węzłów, do sprawdzania;
- **closedSet** - zbiór węzłów, które już odwiedziliśmy i nie będą one ponownie sprawdzane;
- **priorityQueue** - kolejka węzłów, posortowanych malejąco względem całkowitego kosztu;

3. Instrukcja obsługi programu

3.1. Parametry wejściowe

Program obsługiwany jest z konsoli i wspiera definiowanie rodzaju strategii przeszukania grafu, wyboru heurystyki oraz metody wprowadzenia danych:

1. **Algorytmy przeszukania:**
 - a) **-b** dla BFS
 - b) **-d** dla DFS
 - c) **-i** dla iDFS
 - d) **-a** dla A*
2. **Argumenty dla strategii:**
 - a) dla strategii **BFS,DFS,IDFS** należy zdefiniować porządek generowania sąsiadów
 - i. **R** dla losowego porządku
 - ii. **LPGD** w dowolnej permutacji - stały porządek
 - b) dla strategii **A*** należy wybrać jedną z dwóch dostępnych heurystyk:
 - i. **1** dla **ManhattanDistance**
 - ii. **2** dla **InvalidCount**
3. **Wprowadzanie danych:**
 - a) **-f cmd** - stan wejściowy wprowadzany z konsoli
 - b) **-f file={plik z wejściowym węzłem}**
 - a) Program do wizualizacji rozwiązania: Zadaniem programu jest uwidocznienie rozwiązania układanki. Jego parametrami są dwa pliki: output.txt - który obrazuje początkowe ustawienie kafelków oraz solution.txt który w pierwszej linii zawiera informacje o ilości ruchów. W następnej linii zawiera ciąg znaków złożonych z liter 'L', 'P', 'G', 'D' oznaczających kolejno Lewo, Prawo, Góra, Dół

3.2. Przykładowe użycia

- **-a 1 -f cmd** - A* z ManhattanDistance, węzeł startowy z konsoli,
- **-i R -f file=./input.txt** - iDFS z losowym porządkiem, węzeł startowy ładowany z pliku

4. Krótki opis zaimplementowanych klas

Klasy programu rozwiązującego układankę:

1. IDFSStrategy - klasa implementująca rozwiązanie zagadki przy użyciu algorytmu iDFS
2. DFSStrategy - klasa implementująca rozwiązanie zagadki przy użyciu algorytmu DFS
3. AStarPuzzleStrategy - klasa implementująca rozwiązanie zagadki przy użyciu algorytmu A*
4. BFSPuzzleStrategy - klasa implementująca rozwiązanie zagadki przy użyciu algorytmu BFS

Klasy programu wizualizującego uzyskane rozwiązanie

1.

5. Część badawcza

5.1. iDFS

Statystyki przeszukiwania metodą iDFS dla układanek z rozwiązaniem w odległościach 0-7

Kolejność przeszukiwania:LP GD Odległość rozwiązania: 1

- Średnia długość rozwiązania: 1.0
- Średnia ilość odwiedzonych stanów: 4.0
- Średnia maksymalna głębokość rekursji: 2.0
- Średnia ilość wygenerowanych stanów: 6.5

Odległość rozwiązania: 2

- Średnia długość rozwiązania: 2.0
- Średnia ilość odwiedzonych stanów: 15.75
- Średnia maksymalna głębokość rekursji: 3.0
- Średnia ilość wygenerowanych stanów: 15.0

Odległość rozwiązania: 3

- Średnia długość rozwiązania: 3.0
- Średnia ilość odwiedzonych stanów: 36.8
- Średnia maksymalna głębokość rekursji: 4.0
- Średnia ilość wygenerowanych stanów: 33.0

Odległość rozwiązania: 4

- Średnia długość rozwiązania: 4.0
- Średnia ilość odwiedzonych stanów: 69.04166666666667
- Średnia maksymalna głębokość rekursji: 5.0
- Średnia ilość wygenerowanych stanów: 47.916666666666664

Odległość rozwiązania: 5

- Średnia długość rozwiązania: 5.0
- Średnia ilość odwiedzonych stanów: 168.57407407407408
- Średnia maksymalna głębokość rekursji: 6.0
- Średnia ilość wygenerowanych stanów: 146.35185185185185

Odległość rozwiązania: 6

- Średnia długość rozwiązania: 6.0
- Średnia ilość odwiedzonych stanów: 351.8878504672897
- Średnia maksymalna głębokość rekursji: 7.0
- Średnia ilość wygenerowanych stanów: 270.0841121495327

Kolejność przeszukiwania:DLPG Odległość rozwiązania: 1

- Średnia długość rozwiązania: 1.0
- Średnia ilość odwiedzonych stanów: 4.5
- Średnia maksymalna głębokość rekursji: 2.0
- Średnia ilość wygenerowanych stanów: 7.5

Odległość rozwiązania: 2

- Średnia długość rozwiązania: 2.0
- Średnia ilość odwiedzonych stanów: 17.25
- Średnia maksymalna głębokość rekursji: 3.0

- Średnia ilość wygenerowanych stanów: 19.25
- Odległość rozwiązania: 3
- Średnia długość rozwiązania: 3.0
- Średnia ilość odwiedzonych stanów: 38.0
- Średnia maksymalna głębokość rekursji: 4.0
- Średnia ilość wygenerowanych stanów: 35.5
- Odległość rozwiązania: 4
- Średnia długość rozwiązania: 4.0
- Średnia ilość odwiedzonych stanów: 70.875
- Średnia maksymalna głębokość rekursji: 5.0
- Średnia ilość wygenerowanych stanów: 51.791666666666664
- Odległość rozwiązania: 5
- Średnia długość rozwiązania: 5.0
- Średnia ilość odwiedzonych stanów: 162.37037037037038
- Średnia maksymalna głębokość rekursji: 6.0
- Średnia ilość wygenerowanych stanów: 133.3148148148148
- Odległość rozwiązania: 6
- Średnia długość rozwiązania: 6.0
- Średnia ilość odwiedzonych stanów: 362.49532710280374
- Średnia maksymalna głębokość rekursji: 7.0
- Średnia ilość wygenerowanych stanów: 292.7943925233645
- Kolejność przeszukiwania: PGDL Odległość rozwiązania: 1
- Średnia długość rozwiązania: 1.0
- Średnia ilość odwiedzonych stanów: 3.5
- Średnia maksymalna głębokość rekursji: 2.0
- Średnia ilość wygenerowanych stanów: 5.5
- Odległość rozwiązania: 2
- Średnia długość rozwiązania: 2.0
- Średnia ilość odwiedzonych stanów: 18.5
- Średnia maksymalna głębokość rekursji: 3.0
- Średnia ilość wygenerowanych stanów: 22.25
- Odległość rozwiązania: 3
- Średnia długość rozwiązania: 3.0
- Średnia ilość odwiedzonych stanów: 34.0
- Średnia maksymalna głębokość rekursji: 4.0
- Średnia ilość wygenerowanych stanów: 29.0
- Odległość rozwiązania: 4
- Średnia długość rozwiązania: 4.0
- Średnia ilość odwiedzonych stanów: 75.91666666666667
- Średnia maksymalna głębokość rekursji: 5.0
- Średnia ilość wygenerowanych stanów: 62.5
- Odległość rozwiązania: 5
- Średnia długość rozwiązania: 5.0
- Średnia ilość odwiedzonych stanów: 161.35185185185185
- Średnia maksymalna głębokość rekursji: 6.0
- Średnia ilość wygenerowanych stanów: 131.14814814814815
- Odległość rozwiązania: 6
- Średnia długość rozwiązania: 6.0

- Średnia ilość odwiedzonych stanów: 372.1869158878505
- Średnia maksymalna głębokość rekursji: 7.0
- Średnia ilość wygenerowanych stanów: 313.41121495327104
- Kolejność przeszukiwania: DGLP Odległość rozwiązania: 1
- Średnia długość rozwiązania: 1.0
- Średnia ilość odwiedzonych stanów: 4.0
- Średnia maksymalna głębokość rekursji: 2.0
- Średnia ilość wygenerowanych stanów: 6.0
- Odległość rozwiązania: 2
- Średnia długość rozwiązania: 2.0
- Średnia ilość odwiedzonych stanów: 16.25
- Średnia maksymalna głębokość rekursji: 3.0
- Średnia ilość wygenerowanych stanów: 16.75
- Odległość rozwiązania: 3
- Średnia długość rozwiązania: 3.0
- Średnia ilość odwiedzonych stanów: 38.0
- Średnia maksymalna głębokość rekursji: 4.0
- Średnia ilość wygenerowanych stanów: 36.6
- Odległość rozwiązania: 4
- Średnia długość rozwiązania: 4.0
- Średnia ilość odwiedzonych stanów: 79.33333333333333
- Średnia maksymalna głębokość rekursji: 5.0
- Średnia ilość wygenerowanych stanów: 69.79166666666667
- Odległość rozwiązania: 5
- Średnia długość rozwiązania: 5.0
- Średnia ilość odwiedzonych stanów: 158.5185185185185
- Średnia maksymalna głębokość rekursji: 6.0
- Średnia ilość wygenerowanych stanów: 124.94444444444444
- Odległość rozwiązania: 6
- Średnia długość rozwiązania: 6.0
- Średnia ilość odwiedzonych stanów: 352.21495327102804
- Średnia maksymalna głębokość rekursji: 7.0
- Średnia ilość wygenerowanych stanów: 271.3738317757009

5.2. BFS

5.3. A* z heurystyką ManhattanDistance

Odległość	Porządek	Układanka	Wynik
1	LPGD	1 2 3 4 5 6 7 8 9 10 11 0 13 14 15 12	— Steps : 2 — Time: 8 ms — Path length: 2 — Start cost: 2.0 — Avg cost: 2.0 — Generated nodes: 2 — Skipped nodes: 0 — Visited nodes: 3 — Changed path: 0 — Queue size: 2 — OSet size: 2 — CSet path: 1
Odległość	Porządek	Układanka	Wynik
1	DLPG	1 2 3 4 5 6 7 8 9 10 11 0 13 14 15 12	— Steps : 2 — Time: 0 ms — Path length: 2 — Start cost: 2.0 — Avg cost: 2.0 — Generated nodes: 2 — Skipped nodes: 0 — Visited nodes: 3 — Changed path: 0 — Queue size: 2 — OSet size: 2 — CSet path: 1

Odległość	Porządek	Układanka	Wynik
1	PGDL	1 2 3 4 5 6 7 8 9 10 11 0 13 14 15 12	— Steps : 2 — Time: 0 ms — Path length: 2 — Start cost: 2.0 — Avg cost: 2.0 — Generated nodes: 2 — Skipped nodes: 0 — Visited nodes: 3 — Changed path: 0 — Queue size: 2 — OSet size: 2 — CSet path: 1
Odległość	Porządek	Układanka	Wynik
1	DGLP	1 2 3 4 5 6 7 8 9 10 11 0 13 14 15 12	— Steps : 2 — Time: 0 ms — Path length: 2 — Start cost: 2.0 — Avg cost: 2.0 — Generated nodes: 2 — Skipped nodes: 0 — Visited nodes: 3 — Changed path: 0 — Queue size: 2 — OSet size: 2 — CSet path: 1

Odległość	Porządek	Układanka	Wynik
2	LPGD	1 2 3 4 5 6 7 8 9 10 0 12 13 14 11 15	— Steps : 3 — Time: 2 ms — Path length: 3 — Start cost: 4.0 — Avg cost: 4.0 — Generated nodes: 5 — Skipped nodes: 1 — Visited nodes: 6 — Changed path: 0 — Queue size: 4 — OSet size: 4 — CSet path: 2
Odległość	Porządek	Układanka	Wynik
2	DLPG	1 2 3 4 5 6 7 8 9 10 0 12 13 14 11 15	— Steps : 3 — Time: 1 ms — Path length: 3 — Start cost: 4.0 — Avg cost: 4.0 — Generated nodes: 5 — Skipped nodes: 1 — Visited nodes: 6 — Changed path: 0 — Queue size: 4 — OSet size: 4 — CSet path: 2

Odległość	Porządek	Układanka	Wynik
2	PGDL	1 2 3 4 5 6 7 8 9 10 0 12 13 14 11 15	— Steps : 3 — Time: 2 ms — Path length: 3 — Start cost: 4.0 — Avg cost: 4.0 — Generated nodes: 5 — Skipped nodes: 1 — Visited nodes: 6 — Changed path: 0 — Queue size: 4 — OSet size: 4 — CSet path: 2
Odległość	Porządek	Układanka	Wynik
2	DGLP	1 2 3 4 5 6 7 8 9 10 0 12 13 14 11 15	— Steps : 3 — Time: 2 ms — Path length: 3 — Start cost: 4.0 — Avg cost: 4.0 — Generated nodes: 5 — Skipped nodes: 1 — Visited nodes: 6 — Changed path: 0 — Queue size: 4 — OSet size: 4 — CSet path: 2

Odległość	Porządek	Układanka	Wynik
3	LPGD	1 2 3 4 5 6 7 8 9 10 12 0 13 14 11 15	— Steps : 5 — Time: 10 ms — Path length: 4 — Start cost: 4.0 — Avg cost: 6.0 — Generated nodes: 8 — Skipped nodes: 3 — Visited nodes: 9 — Changed path: 0 — Queue size: 5 — OSet size: 5 — CSet path: 4
Odległość	Porządek	Układanka	Wynik
3	DLPG	1 2 3 4 5 6 7 8 9 10 12 0 13 14 11 15	— Steps : 5 — Time: 3 ms — Path length: 4 — Start cost: 4.0 — Avg cost: 6.0 — Generated nodes: 8 — Skipped nodes: 3 — Visited nodes: 9 — Changed path: 0 — Queue size: 5 — OSet size: 5 — CSet path: 4

Odległość	Porządek	Układanka	Wynik
3	PGDL	1 2 3 4 5 6 7 8 9 10 12 0 13 14 11 15	— Steps : 5 — Time: 3 ms — Path length: 4 — Start cost: 4.0 — Avg cost: 6.0 — Generated nodes: 8 — Skipped nodes: 3 — Visited nodes: 9 — Changed path: 0 — Queue size: 5 — OSet size: 5 — CSet path: 4
Odległość	Porządek	Układanka	Wynik
3	DGLP	1 2 3 4 5 6 7 8 9 10 12 0 13 14 11 15	— Steps : 5 — Time: 3 ms — Path length: 4 — Start cost: 4.0 — Avg cost: 6.0 — Generated nodes: 8 — Skipped nodes: 3 — Visited nodes: 9 — Changed path: 0 — Queue size: 5 — OSet size: 5 — CSet path: 4

Odległość	Porządek	Układanka	Wynik
4	LPGD	1 2 3 4 5 0 7 8 9 6 11 12 13 10 14 15	— Steps : 5 — Time: 3 ms — Path length: 5 — Start cost: 8.0 — Avg cost: 8.0 — Generated nodes: 10 — Skipped nodes: 3 — Visited nodes: 11 — Changed path: 0 — Queue size: 7 — OSet size: 7 — CSet path: 4
Odległość	Porządek	Układanka	Wynik
4	DLPG	1 2 3 4 5 0 7 8 9 6 11 12 13 10 14 15	— Steps : 5 — Time: 3 ms — Path length: 5 — Start cost: 8.0 — Avg cost: 8.0 — Generated nodes: 10 — Skipped nodes: 3 — Visited nodes: 11 — Changed path: 0 — Queue size: 7 — OSet size: 7 — CSet path: 4

Odległość	Porządek	Układanka	Wynik
4	PGDL	1 2 3 4 5 0 7 8 9 6 11 12 13 10 14 15	— Steps : 5 — Time: 3 ms — Path length: 5 — Start cost: 8.0 — Avg cost: 8.0 — Generated nodes: 10 — Skipped nodes: 3 — Visited nodes: 11 — Changed path: 0 — Queue size: 7 — OSet size: 7 — CSet path: 4
Odległość	Porządek	Układanka	Wynik
4	DGLP	1 2 3 4 5 0 7 8 9 6 11 12 13 10 14 15	— Steps : 5 — Time: 3 ms — Path length: 5 — Start cost: 8.0 — Avg cost: 8.0 — Generated nodes: 10 — Skipped nodes: 3 — Visited nodes: 11 — Changed path: 0 — Queue size: 7 — OSet size: 7 — CSet path: 4

Odległość	Porządek	Układanka	Wynik
5	LPGD	1 2 3 4 5 6 11 7 9 10 8 0 13 14 15 12	— Steps : 10 — Time: 15 ms — Path length: 6 — Start cost: 6.0 — Avg cost: 10.0 — Generated nodes: 19 — Skipped nodes: 8 — Visited nodes: 20 — Changed path: 0 — Queue size: 11 — OSet size: 11 — CSet path: 9
Odległość	Porządek	Układanka	Wynik
5	DLPG	1 2 3 4 5 6 11 7 9 10 8 0 13 14 15 12	— Steps : 10 — Time: 16 ms — Path length: 6 — Start cost: 6.0 — Avg cost: 10.0 — Generated nodes: 19 — Skipped nodes: 8 — Visited nodes: 20 — Changed path: 0 — Queue size: 11 — OSet size: 11 — CSet path: 9

Odległość	Porządek	Układanka	Wynik
5	PGDL	1 2 3 4 5 6 11 7 9 10 8 0 13 14 15 12	— Steps : 10 — Time: 16 ms — Path length: 6 — Start cost: 6.0 — Avg cost: 10.0 — Generated nodes: 19 — Skipped nodes: 8 — Visited nodes: 20 — Changed path: 0 — Queue size: 11 — OSet size: 11 — CSet path: 9
Odległość	Porządek	Układanka	Wynik
5	DGLP	1 2 3 4 5 6 11 7 9 10 8 0 13 14 15 12	— Steps : 10 — Time: 11 ms — Path length: 6 — Start cost: 6.0 — Avg cost: 10.0 — Generated nodes: 19 — Skipped nodes: 8 — Visited nodes: 20 — Changed path: 0 — Queue size: 11 — OSet size: 11 — CSet path: 9

Odległość	Porządek	Układanka	Wynik
6	LPGD	1 2 3 4 5 6 7 8 9 11 14 12 13 0 10 15	— Steps : 15 — Time: 18 ms — Path length: 7 — Start cost: 8.0 — Avg cost: 12.0 — Generated nodes: 29 — Skipped nodes: 13 — Visited nodes: 30 — Changed path: 0 — Queue size: 16 — OSet size: 16 — CSet path: 14
Odległość	Porządek	Układanka	Wynik
6	DLPG	1 2 3 4 5 6 7 8 9 11 14 12 13 0 10 15	— Steps : 15 — Time: 23 ms — Path length: 7 — Start cost: 8.0 — Avg cost: 12.0 — Generated nodes: 29 — Skipped nodes: 13 — Visited nodes: 30 — Changed path: 0 — Queue size: 16 — OSet size: 16 — CSet path: 14

Odległość	Porządek	Układanka	Wynik
6	PGDL	1 2 3 4 5 6 7 8 9 11 14 12 13 0 10 15	— Steps : 15 — Time: 21 ms — Path length: 7 — Start cost: 8.0 — Avg cost: 12.0 — Generated nodes: 29 — Skipped nodes: 13 — Visited nodes: 30 — Changed path: 0 — Queue size: 16 — OSet size: 16 — CSet path: 14
Odległość	Porządek	Układanka	Wynik
6	DGLP	1 2 3 4 5 6 11 7 9 10 8 0 13 14 15 12	Brak rozwiązania
Odległość	Porządek	Układanka	Wynik
7/1	R	1 2 7 0 5 6 4 3 9 10 11 8 13 14 15 12	— Steps : 126 — Time: 89 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 261 — Skipped nodes: 124 — Visited nodes: 262 — Changed path: 0 — Queue size: 137 — OSet size: 137 — CSet path: 125

Odległość	Porządek	Układanka	Wynik
7/2	R	1 2 7 3 5 6 4 8 9 10 11 0 13 14 15 12	— Steps : 218 — Time: 125 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 440 — Skipped nodes: 218 — Visited nodes: 441 — Changed path: 0 — Queue size: 224 — OSet size: 224 — CSet path: 217
Odległość	Porządek	Układanka	Wynik
7/3	R	1 2 7 3 5 6 11 4 9 10 8 0 13 14 15 12	— Steps : 260 — Time: 118 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 521 — Skipped nodes: 260 — Visited nodes: 522 — Changed path: 0 — Queue size: 263 — OSet size: 263 — CSet path: 259

Odległość	Porządek	Układanka	Wynik
7/4	R	1 2 7 3 5 6 11 4 9 10 15 8 13 14 0 12	— Steps : 226 — Time: 93 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 461 — Skipped nodes: 225 — Visited nodes: 462 — Changed path: 0 — Queue size: 237 — OSet size: 237 — CSet path: 225
Odległość	Porządek	Układanka	Wynik
7/5	R	1 2 7 3 5 6 11 4 9 0 10 8 13 14 15 12	— Steps : 353 — Time: 147 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 713 — Skipped nodes: 356 — Visited nodes: 714 — Changed path: 0 — Queue size: 362 — OSet size: 362 — CSet path: 352

Odległość	Porządek	Układanka	Wynik
7/6	R	1 0 7 3 5 2 6 4 9 10 11 8 13 14 15 12	— Steps : 250 — Time: 144 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 501 — Skipped nodes: 250 — Visited nodes: 502 — Changed path: 0 — Queue size: 253 — OSet size: 253 — CSet path: 249
Odległość	Porządek	Układanka	Wynik
7/7	R	1 2 7 3 5 10 6 4 9 0 11 8 13 14 15 12	— Steps : 231 — Time: 73 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 493 — Skipped nodes: 230 — Visited nodes: 494 — Changed path: 0 — Queue size: 264 — OSet size: 264 — CSet path: 230

Odległość	Porządek	Układanka	Wynik
7/8	R	1 2 7 3 0 5 6 4 9 10 11 8 13 14 15 12	— Steps : 258 — Time: 163 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 517 — Skipped nodes: 258 — Visited nodes: 518 — Changed path: 0 — Queue size: 261 — OSet size: 261 — CSet path: 257

Odległość	Porządek	Układanka	Wynik
7/9	R	1 6 2 3 5 7 0 4 9 10 11 8 13 14 15 12	— Steps : 297 — Time: 131 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 611 — Skipped nodes: 298 — Visited nodes: 612 — Changed path: 0 — Queue size: 316 — OSet size: 316 — CSet path: 296
Odległość	Porządek	Układanka	Wynik
7/10	R	1 6 2 3 5 10 7 4 9 0 11 8 13 14 15 12	— Steps : 301 — Time: 156 ms — Path length: 8 — Start cost: 1.0 — Avg cost: 7.0 — Generated nodes: 618 — Skipped nodes: 303 — Visited nodes: 619 — Changed path: 0 — Queue size: 319 — OSet size: 319 — CSet path: 300

5.4. A* oraz porównanie heurystyk

```

1    2    0    3
5    6    7    8
9    10   4    11
13   14   15   12

```

Manhattan:

- Steps : 81
- Time: 54 ms
- Path length: 11
- Start cost: 10.0
- Avg cost: 20.0
- Generated nodes: 167
- Skipped nodes: 79

- Visited nodes: 168
- Changed path: 0
- Queue size: 88
- OSet size: 88
- CSet path: 88

InvalidCount:

- Steps : 2386
- Time: 3293 ms
- Path length: 11
- Start cost: 1.0
- Avg cost: 10.0
- Generated nodes: 4690
- Skipped nodes: 2430
- Visited nodes: 4691
- Changed path: 0
- Queue size: 2306
- OSet size: 2306
- CSet path: 2305

6. Wnioski

Literatura

- [1] http://en.wikipedia.org/wiki/Depth-first_search
- [2] http://en.wikipedia.org/wiki/Iterative_deepening_depth-first_search